PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 7 : G06F 11/30, 12/14, H04L 9/00, 9/32

A1

(11) International Publication Number: WO 00/34867

(43) International Publication Date: 15 June 2000 (15.06.00)

(21) International Application Number: PCT/US99/28717

(22) International Filing Date: 3 December 1999 (03.12.99)

(30) Priority Data:

60/111,638	9 December 1998 (09.12.98)	US
09/447,500	23 November 1999 (23.11.99)	US

(71) Applicant *(for all designated States except US)*: NETWORK ICE CORPORATION [US/US]; 30 West 39th Street, Suite 103, San Mateo, CA 94403 (US).

(72) Inventor; and

(75) Inventor/Applicant *(for US only)*: GRAHAM, Robert, D. [US/US]; 350 Sharon Park Drive, #H303, Menlo Park, CA 94025 (US).

(74) Agents: MILLIKEN, Darren, J. et al.; Blakely, Sokoloff, Taylor & Zafman LLP, 12400 Wilshire Boulevard, 7th Floor, Los Angeles, CA 90025 (US).

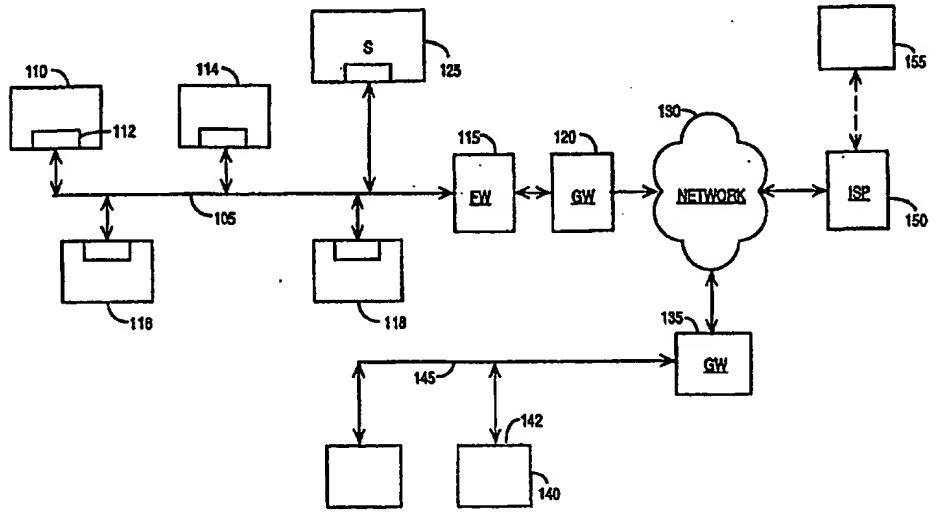
(81) Designated States: AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, DE, DK, DM, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published

With international search report.

Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: A METHOD AND APPARATUS FOR PROVIDING NETWORK AND COMPUTER SYSTEM SECURITY



(57) Abstract

A network intrusion detection and response system (115) and method is disclosed for detecting and preventing misuse of network resources. More particularly, the system and method dynamically self-adjusts to changes in network activity using a plurality of alert levels wherein each successively higher alert level triggers a corresponding heightened security response from the networked computer (125) being misused. These heightened alert levels are integrated on both the system (individual node) and the network level.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece			TR	Turkey
BG	Bulgaria	HU	Hungary	ML	Mali	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MN	Mongolia	UA	Ukraine
BR	Brazil	IL	Israel	MR	Mauritania	UG	Uganda
BY	Belarus	IS	Iceland	MW	Malawi	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	NZ	New Zealand		
CM	Cameroon			PL	Poland		
CN	China	KR	Republic of Korea	PT	Portugal		
CU	Cuba	KZ	Kazakstan	RO	Romania		
CZ	Czech Republic	LC	Saint Lucia	RU	Russian Federation		
DE	Germany	LI	Liechtenstein	SD	Sudan		
DK	Denmark	LK	Sri Lanka	SE	Sweden		
EE	Estonia	LR	Liberia	SG	Singapore		

A METHOD AND APPARATUS FOR PROVIDING NETWORK AND COMPUTER SYSTEM SECURITY

PRIORITY

This application claims the benefit of U.S. Provisional Application No. 60/111,638, filed 12/9/98.

BACKGROUND OF THE INVENTION

Field of the Invention

This invention relates to data network management. More particularly, the invention relates to an improved system and method for detecting and preventing unauthorized use of data network resources.

Description of the Related Art

The rapid increase in the use of data networks by both corporations and private organizations has created a need for improved security and network management techniques. Organizations today store substantial amounts of confidential information on network servers and workstations including trade secrets, marketing strategies, financial documents, and classified technical information. The disclosure of such information to the public would, in most instances, cause severe damage to the organization.

In addition to the danger of confidential information being read out from the network, there is also a danger of unwanted information being *written to* the network. For example, with a working knowledge of how to evade currently available security systems, computer hackers (i.e., unauthorized users) are capable of crashing network servers and workstations, corrupting valuable data, and uploading computer viruses to the network. As such, organizations are forced to spend millions of dollars each year in an attempt to prevent this type of data network intrusion.

One system for handling a type of network misuse is commonly referred to as a "firewall." Firewalls are generally situated between a local area network (hereinafter "LAN") and all other external networks (e.g., the Internet). The firewall analyzes all incoming and outgoing digital information and makes a decision as to whether the information should be passed through or discarded. The firewall uses one or more algorithms provided by a network administrator to perform this analysis. For example, a network administrator may configure tables which list acceptable source and destination addresses for network traffic. Traffic addressed to an unlisted source or destination will be filtered out and discarded by the firewall.

Firewalls provide insufficient protection against computer hackers for a variety of reasons. One major reason is that firewalls only protect LANs from the outside world whereas the threat posed by computer hackers is not merely external. In fact, the majority of potential computer hackers are internal computer users, most of whom already have access to the LAN. Although an individual user will typically be provided only limited access to LAN resources, the user may fraudulently acquire access to additional resources by misappropriating other users' passwords (or using other known computer hacking techniques).

A second problem associated with firewalls is that they are static in nature, requiring continuous updates by network administrators to work properly. If a computer hacker obtains the information necessary to break through the firewall (i.e., information needed to disguise his data as data originating from a legitimate source) he will acquire access to resources on the LAN. Another significant problem with firewalls is that they exclude data in an overly simplistic fashion: data is either passed through or it is discarded. No additional analysis is performed on incoming or outgoing data to determine whether the originator of the data – who may be disguising himself to the firewall – is attempting to misuse resources on the LAN.

One technique used to augment the limited scope of protection provided by firewalls has been referred to as "misuse detection." Misuse detection is the process of monitoring and reporting unauthorized or inappropriate activity on network computers.

For example, Smaha et al., U.S. Patent No. 5,557,742 (hereinafter referred to as "Smaha") discloses a process for detecting a misuse condition by applying predetermined "misuse signatures" to identify known misuses of networked computers. An example of a misuse signature is four unsuccessful logins on a network computer followed by a successful login (see Smaha column 12, lines 12-13).

One limitation of this type of system is that it (like a firewall) is static in nature and requires continuous updates. Thus, as new "misuse signatures" are discovered, they must continually be incorporated into the detection system by programmers or network administrators. Requiring the manual incorporation of new "misuse signatures" is inefficient and will allow an experienced computer hacker access to network resources until his particular "misuse signature" has been determined.

An additional problem with prior art misuse detection systems such as Smaha is that once a potential misuse condition has been observed, the network is limited in its ability to respond to the condition. For example, once a potential hacker has unsuccessfully attempted to login to a networked computer four times as described above, the networked computer will simply deny the hacker access to its resources, rather than taking steps to acquire additional information about the hacker and warn other computers on the network about the hacker. Conversely, if three unsuccessful logins are detected rather than four (followed by a successful login) Smaha discloses no mechanism to raise the system to a heightened security level (i.e., an intermediate level) wherein additional information is collected from the potential hacker before providing access to the network computer.

An additional problem with prior art misuse detection systems such as Smaha is that automated systems can only identify activity as being suspicious, but cannot conclusively differentiate among deliberate misuse attempts, accidents (e.g., user enters the wrong password), or normal incidents (e.g., network manager uses pings to monitor network performance). Thus, prior art misuse detection systems record all suspicious events and rely upon the intelligence of the operator to wade through the "false-positives" in order to find salient records.

SUMMARY OF THE INVENTION

An article of manufacture is disclosed including a sequence of instructions stored on a computer-readable media which when executed by a network node cause the network node to perform the acts of: modifying an alert variable based on data transmissions originating from a suspect node; triggering a first response when said alert variable reaches a first predetermined threshold level; and
triggering a second response when said alert variable reaches a second predetermined threshold level.

BRIEF DESCRIPTION OF THE DRAWINGS

A better understanding of the present invention can be obtained from the following detailed description in conjunction with the following drawings, in which:

FIG. 1 illustrates generally two local area networks and an Internet Service Provider ("ISP") communicating through a larger network according to one embodiment of the invention.

FIG. 2A illustrates portions of the OSI protocol stack according to one embodiment of the invention.

FIG. 2B illustrates an analysis and filter system implemented between the data link and network layers of the OSI stack according to one embodiment of the invention.

FIG. 3 is a graph illustrating the aggravation response according to one embodiment of the invention.

FIG. 4 is a graph illustrating the overall target aggravation response and the suspect-specific target aggravation response according to one embodiment of the invention.

FIG. 5 is a graph illustrating the overall target aggravation response and the suspect-specific target aggravation response according to another embodiment of the invention.

FIG. 6 is a graph illustrating the suspect-specific network aggravation response of another embodiment of the invention.

FIG. 7 is a graph illustrating the operation of event overlap according to one embodiment of the invention.

FIG. 8 is a graph illustrating an event data object according to one embodiment of the invention.

FIG. 9 illustrates two time periods which are used to calculate difference reports.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

What is needed is an improved system and method to aid network administrators in detecting and preventing misuse of network resources. More particularly, what is needed is a comprehensive intrusion detection and response system which will dynamically self-adjust to changes in network activity. What is also needed is an intrusion detection and response system which implements a plurality of alert levels wherein each successively higher alert level triggers a corresponding heightened security response from the networked computer being misused. What is also needed is a system which integrates such heightened alert levels on both the system and the network level. What is also needed is an intrusion detection and response system which can be implemented at low cost and using currently-existing hardware and software (i.e., network computers). What is also needed is an intrusion detection and response system which aggressively filters false-positives while still recording the necessary information to identify an attack.

General Network Configuration

Figure 1 generally depicts a local area network (hereinafter "LAN") 105 over which a plurality of nodes (e.g., 110, 125) communicate. Nodes on LAN 105 include a server 125, a plurality of workstations 110, 114, 116, and 118, and a firewall 115. LAN 105 communicates over a larger network 130 (e.g., the Internet) through a gateway 120. The gateway 120 in one embodiment translates between different network protocols of LAN 105 and network 130 as necessary.

A second LAN 145 and a plurality of nodes 140 are illustrated communicating over network 130 through a second gateway 135. An Internet Service Provider 150 (hereinafter "ISP") is shown communicating across network 130. Node 155 may communicates with ISP 150 in a variety of ways including, for example, a standard telephone connection or an ISDN or cable modem connection.

Each workstation 110, 114, 116, and 118 and server 125 on LAN 105 is a computer comprising a processor and a memory with which software implementing the functionality of the intrusion detection system described herein is executed. Such a computer system stores and communicates (internally or with other computer systems over a network) code and data using machine readable media, such as magnetic disks, random access memory, read only memory, carrier waves, signals, etc. In addition, while one embodiment is described in which the parts of the present invention are implemented in software, alternative embodiments can implement one or more of these parts using any combination of software, firmware and/or hardware.

Each workstation 110, 114, 116, and 118 and server 125 also includes a network interface 112 comprised of hardware and/or software which allows each of the networked computers 110 114, 116, and 118 and server 125 to transmit data over LAN 105. In one embodiment, the hardware portion of network interface 112 is a network interface card connected to the I/O bus of the networked computer 110.

Aggravation Levels

A. Target Aggravation

As used herein, an "incident" is the receipt of a particular type of data transmission by a network node (e.g., a workstation 110, 114, 116, 118, server 125, firewall 115, gateway 120, ISP 150, etc.) hereinafter referred to as the "target node" or "target," originating from another network node, hereinafter referred to as the "suspect node" or "suspect." Certain types of incidents are inherently suspicious from a system security point of view. For example, the transmission of an invalid user ID or password from a suspect node to a target node may indicate that an unauthorized user is attempting to gain access to the target node. Moreover, a *series* of login failures over a relatively short period of time creates a substantially greater suspicion of the suspect node than a single login failure. In other words, a single login failure will generally be the result of an authorized user typing his user ID or password incorrectly from the suspect node. However, with each successive invalid login attempt it becomes substantially less likely that the user of the suspect node is simply making a mistake, and more likely that the user is attempting to gain unauthorized access to the target node.

To address these concerns, in one embodiment of the present system a self-modifying alert level is provided for each target node. The alert level, referred to herein as an "aggravation level," changes dynamically in response to incidents arriving at the target node and to the history of incidents which have previously arrived at the target node. The target node may react to new incidents differently depending on its current aggravation level.

For example, **figure 3** is a graph which shows how the aggravation level of a target node in one embodiment of the present system passes through a plurality of threshold levels 310-350. For the purposes of the following discussion, as time increases in **figure 3** (to the right along the x-axis) it is assumed that the target node receives continual invalid login incidents from one or more suspect nodes. Each threshold is a trigger to take some action. For example, as the aggravation level increases, as shown in curve 360, the first aggravation threshold reached is threshold 310. In one embodiment, at threshold 310 the target node begins a passive scan on all incoming incidents. For example, the target node may begin recording concurrent incoming incidents in an incident log file so that the full extent of the intrusion can be identified. Examples of incidents include web server "hits" (file access), mail commands, port scans, and pings from the suspect(s) to the target.

At the next aggravation threshold, threshold 320, the target node of one embodiment will begin actively scanning the suspect nodes causing the incidents, in an attempt to acquire identification information about the suspect nodes. One example of an active scan is the "finger" command. The finger command returns, among other information, the suspect nodes' network addresses and the identity of the users who are currently logged in to the suspect nodes. At this aggravation threshold, the target may also increase its passive scanning for new incidents. Thus, at aggravation threshold 320, the target may begin to actively acquire information about the suspects and also may increase the logging associated with new incidents.

As the target continues to receive login failures from one or more suspects, its aggravation level reaches threshold 330. Here, the target of one embodiment begins a

more aggressive active scan of the suspects. For example, using the "traceroute" command the target may attempt to ascertain the complete network route taken by data originating form the suspect. Thus, referring once again to **figure 1**, if the target node is node 110 on LAN 105 and the suspect node is node 140 on LAN 145, the traceroute command will trace the communication path between the suspect and the target through network 130 (i.e., it will report the network addresses of all the routers in between). In addition, the target may query the suspect's local gateway – gateway 135 in the present example – for the suspect's data link address (a.k.a. media access control address) using the Simple Network Management Protocol ("SNMP"). More specifically, identification via SNMP can consist of a "get-next" sweep of the table of the ifEntry field (usually, one entry per interface).

As the target's aggravation reaches threshold 340, the target in one embodiment will take additional steps to ensure that an unauthorized suspect is not provided with access to its resources. At this threshold level the target may require additional authentication information from suspects before providing the suspects access. For example, when a suspect transmits *correct* login information (i.e. the correct user ID and password) to the target, the target may initially return an "invalid login" response back to the suspect. Only if the suspect once again transmits the correct login information will the target provide the suspect access. By forcing a double logon in this manner, the target will prevent suspects from acquiring unauthorized access by using automated login scripts, i.e., scripts which run through a series of user ID's and/or passwords in an attempt to determine a valid combination.

At its highest aggravation threshold, threshold 350, the target has received numerous successive login failure incidents, resulting in an unreasonably high probability that an unauthorized suspect is attempting to gain access to its resources. Therefore, at this aggravation level the target may take the final step of blocking incoming incidents (e.g., from every one, from everyone outside its LAN, from a given set of suspects, etc.), thereby denying access to its resources to all of the suspects. The target may also decide to take active scanning measures beyond simple identification, such sending a "ping-of-death" in an attempt to shut down the suspect(s).

Throughout the preceding discussion, specific target node reactions at predetermined aggravation threshold levels have been described. However, alternative embodiments of the invention can use higher, lower and different types of thresholds, threshold reactions, etc., without departing from the underlying principles of the present invention. For example, while the embodiment described above uses the "finger" command at threshold 320 and the "traceroute" command at threshold 330, any alternative/additional active scanning techniques (e.g., "identd" or "ping" commands) could be implemented at these stages without departing from the scope of the present invention. Moreover, different combinations of target node reactions could be employed as well. For example, additional passive scanning techniques could be implemented at each of the threshold levels 310-340 in conjunction with active scanning techniques such as complete packet capture that logs every byte of traffic from the suspect(s).

Moreover, the various threshold levels described above may be calculated using different mathematical equations. For example, the when a new incident arrives at the target node, the aggravation value associated with the incident "A_i" may simply be added to the current aggravation of the target "A" such that a threshold will trigger only if $A_i + A > T$. Alternatively, an equation such as $.5\ A_i + 2A > .75T$ would take half of the incident aggravation level added to twice the current target aggravation level and compare the result with three-quarters of the threshold level.

In addition, as shown by curve 370 of **figure 3**, different types of incidents may provoke different target reactions over time. For example, while the target aggravation level may remain above a maximum threshold value 350 during a continuous series of login failures, it may decrease over time during, for example, for a continuous series of "pings." A "ping," short for "Packet Internet Groper," is a command used to determine whether a node at a particular network address is online. Thus, it is frequently used to test and debug a network by sending out a packet and waiting for a response. In one embodiment of the present system, a series of pings directed at a target will initially increase the target's aggravation level because a suspect may ping a target to verify the target's network address before attempting to evade or disable the target's security system.

Accordingly, as shown in curve 370, the target may initially react with the same aggravation as if it were receiving login failures (i.e.; curves 360 and 370 initially overlap). However, over time the aggravation level may decrease, even if the target continues to receive pings. This is because as time passes it becomes less and less likely that the suspect pinging the target is attempting to gain unauthorized access to the target. For example, the suspect may be running network management software and may simply be monitoring the response time of the target node by sending out a ping every few minutes. Alternatively, a network administrator may ping various nodes on the network to troubleshoot portions of the network. As such, when a suspect pings a target over an extended period of time, it is more likely that the suspect is pinging the target for legitimate purposes.

Furthermore, while curves 360 and 370 illustrate an example when the aggravation level is changed at the same rate for different incident types, alternative embodiments vary the rate of change based on the type of incident.

B. Network Aggravation

In addition to maintaining an aggravation level for each individual target node on the network, an aggravation level may also be maintained for the entire network (hereinafter referred to as the "network aggravation level"). For example, the system may be configured such that server 125 receives each target's current aggravation level. Particularly, server 125 may be configured to query each of the target nodes 110, 114, 116, and 118 on LAN 105 at predetermined time intervals to record each target's current aggravation level. Alternatively, or in addition, each target may actively communicate its current aggravation level to server 125 (without first receiving a request from server 125) at predetermined time intervals or when the target's aggravation level transitions from one threshold to another.

The network aggravation level may be calculated by simply averaging the individual target aggravation levels. Alternatively, it may bear some other mathematical relationship with the target aggravation levels. For example, it may be configured to

respond only to significant short term changes in individual target aggravations (which could indicate that a suspect is initiating an attack on a target).

Regardless of how the network aggravation level is calculated, however, it can be *used* in a number of different ways. For example, a first network aggravation threshold may force all target nodes on LAN 105 to require double logins (as described above). A second network aggravation threshold may be set to block network traffic. For example, if the network aggravation level reaches a particular threshold, server 125 may direct gateway 120 to disable all incoming traffic. In this embodiment, server 125 may be configured to differentiate between target aggravations resulting from internal suspects (i.e., residing on LAN 105) and external suspects (i.e., residing over network 130). To accomplish this, server 125 may maintain an internal network aggravation level and an external network aggravation level. The server in this embodiment would disable data traffic through gateway 120 only if the external suspect aggravation level reached a particular threshold.

C. Suspect-Specific and Overall Target Aggravation

In another embodiment, each target node (i.e., workstation 110 and/or server 125) on LAN 105 maintains a unique aggravation level for each suspect node with which it communicates (i.e., from which incidents have originated). This aggravation level will be referred to herein as the "suspect-specific aggravation." In addition to maintaining a suspect-specific aggravation for each individual suspect node, the target node may also maintain an overall aggravation level, referred to herein as the "target aggravation" or the "overall target aggravation."

Although a correlation exists between the various suspect-specific aggravation levels and the overall target aggravation level during certain periods of time, this correlation does not always exist. More specifically, the overall target aggravation will react primarily to new incidents, whereas the suspect-specific aggravation will react to both new and old incidents.

The relationship between the two different aggravations for one embodiment will now be described with reference to **figure 4**. As shown in **figure 4**, if at time t=0 suspect A begins a series of unsuccessful login attempts to the target node, the target's aggravation level towards suspect A (shown as curve 410) will increase with each unsuccessful attempt. The target's suspect-specific aggravation for suspect A will pass through one or more thresholds 450-455 (as described above with reference to **figure 3**) and will level out at some maximum aggravation value 460 representing 100% aggravation. The target's aggravation towards suspect A will remain at this level as long as continuous login failures are received from suspect A. In fact, even if the login failures from suspect A cease, the suspect-specific aggravation in one embodiment will remain at 100% until the target is reset by a network administrator or an automated reset mechanism.

Also illustrated in **figure 4** is curve 420 representing the overall target aggravation. The overall target aggravation, representing the target's aggravation *in general*, reacts to incoming incidents from *all* suspects. Moreover, the overall target aggravation thresholds (e.g., 450, 454, and 455) may cause different reaction types directed towards all suspects (rather than towards a particular suspect as with the suspect-specific thresholds). Thus, as suspect A continually attempts unsuccessfully to log in to the target, the overall target aggravation initially increases along with its suspect-specific aggravation for suspect A. Thus, when the overall target aggravation (curve 420) reaches aggravation threshold 450, the target may conduct a passive scan of *all* incoming incidents (rather than merely scanning incidents originating from a particular suspect). Likewise, when the overall target aggravation reaches threshold 454 the target may require increased authentication from all suspects and at threshold 455 the target may block communication with all suspects.

Thus, at 470 when a second suspect – suspect B – initially pings the target node, the target node may not respond to the ping because the target's *overall* aggravation level is higher than threshold 455 (i.e., the target is blocking communication with all suspects). Note that the suspect-specific aggravation level for suspect B 430 is low at this point because this is the first time that suspect B has pinged the target.

Eventually, however, the overall target aggravation level begins to decrease. As stated above, the overall target aggravation is affected more by incidents which have not been observed continually over a long period of time, whereas the suspect-specific aggravation reacts to both long term and short-term incidents. The logic behind this behavior is that when the target is being aggravated by only a single suspect (or a few suspects) over an extended period of time, there is no need for the target to be suspicious of *all* suspects – only the ones causing its aggravation.

Moreover, because the target's suspect-specific aggravation towards suspect A it beyond threshold 455, the target has taken steps to block communication with suspect A. The target, therefore, no longer needs to maintain a heightened overall target aggravation because the one suspect causing the login failures has been blocked out.

The second time that suspect B pings the target (at 480), the overall target aggravation is low enough so that the target may now respond to the ping. As shown in curves 430 and 440, both of suspect B's pings increase the target's suspect-specific aggravation towards suspect B and the overall target aggravation level 420. The increase in aggravation level may be related to the target aggravation level: incident 470 may be more aggravating than incident 480 because the system is at a more sensitive state.

Figure 5 is a graph illustrating the overall target aggravation response and the suspect-specific target aggravation response according to another embodiment of the invention. The primary difference between **figure 4** and **figure 5** is in the behavior of the target's suspect-specific aggravation towards suspect A (illustrated as curve 510). Suspect A in this example is continually pinging the target rather than continually causing login failures. Therefore, due to the different implications of receiving a series of pings from a suspect and a receiving a series of invalid logins from a suspect (described above in more detail with respect to **figure 2**) the target's suspect-specific aggravation towards suspect A eventually decreases over time along with the overall target aggravation.

D. Suspect-Specific and Overall Network Aggravation

Just as each individual target in one embodiment maintains an overall target aggravation level and one or more suspect-specific target aggravation levels, server 125 in one embodiment maintains an "overall network aggravation" level and a separate "suspect-specific network aggravation" level.

The graph in figure 6 illustrates how the suspect-specific network aggravation level (curve 600) may react to invalid logins and pings originating from suspect A and directed towards target nodes 110, 114, 116, and 118 on LAN 105. At 605, suspect A pings target 110. As a result, target 110's suspect-specific aggravation towards suspect A increases as shown in curve 610, which initially coincides with the suspect-specific network aggravation level towards suspect A (curve 600). Suspect A's ping causes the suspect-specific network aggravation level to cross threshold 660.

As previously described, threshold 660 may be set to trigger any number of responses from server 125 and/or target nodes 110, 114, 116, and 118. For example, at this threshold server 125 may begin passive scanning of all suspects over LAN 105 or network 130. In another embodiment, server 125 does not conduct any passive scanning of suspect A itself. Rather, when the suspect-specific network aggravation level for a particular suspect reaches a particular threshold which calls for passive scanning, server 125 notifies all target nodes on LAN 105 that they should individually increase their suspect-specific target aggravation towards suspect A.

At 615, suspect A pings target 114 and target 114's suspect-specific aggravation for suspect A increases as shown in curve 620. In response to this second ping by suspect A the suspect-specific network aggravation towards suspect A increases along curve 600 past a second threshold 661. Similarly, at 625, when suspect A unsuccessfully attempts to login to target 116, target 116's suspect-specific target aggravation increases (curve 630), and this incident contributes to the suspect-specific network aggravation towards suspect A. As suspect A continues to ping and/or produce login failures at 635 and 645, the suspect-specific network aggravation also increases as shown in curve 600.

Throughout this process, several additional thresholds 661-664 may be crossed. The same and/or different types of threshold responses described above with reference to the various target aggravation levels can also be implemented for the entire network. Thus, at threshold 661, one or more of the nodes on LAN 105 (e.g., server 125) may begin an active scan of suspect A to acquire additional information about suspect A on behalf of the entire network. At threshold 662, server 125 may begin a more extensive active scan and may also increase its passive scanning of suspect A. At threshold 663, server 125 may require suspect A to provide increased authentication to access any target node on LAN 105 (e.g., double logons). Finally, at the maximum threshold 664, server 125 may block all communication with suspect A over LAN 105. Server 125 may accomplish this either by communicating directly with gateway 120 and firewall 115, or by signaling all targets in LAN 105 to individually block communication with suspect A.

One important feature of the suspect-specific network aggravation response disclosed in **figure 6** is that it increases based on *all* network-wide incidents originating from suspect A. Thus, while the suspect-specific target aggravations towards suspect A (curves 610, 620, 630, 640, and 650) individually remain low, the suspect-specific *network* aggravation response to each of these incidents has a cumulative effect and suspect A is properly identified as a suspicious node.

The suspect-specific network aggravation can be calculated in a variety of ways. In one embodiment, it is merely the sum of all suspect-specific target aggravation levels for suspect A. In another embodiment, it is the *average* of all suspect A suspect-specific target aggravation levels. In still another embodiment, the suspect-specific network aggravation is calculated independently of the individual suspect-specific target aggravation levels. This embodiment might be particularly useful on a LAN with a substantial number of target nodes.

For example, if 250 target nodes reside on LAN 105, suspect A may only make one attempt to gain unauthorized access to the first target and may then move on, making only one attempt at each of the remaining 249 targets. By the time suspect A is ready to make another attempt at the first target, the suspect-specific target aggravation level of

the first target for suspect A might be too low to trigger a threshold due to the time lapse between login failures. However, if a central repository (e.g., server 125) on LAN 105 is following suspect A's activity across the entire network, then the suspect-specific network aggravation level for suspect A should be very high. In fact, if suspect A has attempted unsuccessfully to log in to all 250 targets, then the suspect-specific network aggravation level for suspect A should have surpassed its maximum threshold value, and all network communication with suspect A should be blocked as described above.

Just as each individual target may maintain an "overall target aggravation" which represents the target's aggravation in general, server 125 (or other node on LAN 105) may maintain an "overall network aggravation," which represents the network aggravation in general. Similarly, just as the overall target aggravation responds primarily to new incidents, the "overall network aggravation" in the present embodiment responds primarily to new network-wide incidents.

Thus, as suspect A in the preceding example moves from one target to the next across LAN 105, attempting to log in to each of the 250 individual target nodes, the overall network aggravation level will initially increase along with the suspect-specific network aggravation towards suspect A. However, as time passes and suspect A continues to cause login failures across LAN 105, the overall network aggravation may drop off. The logic here is similar to that for the overall target aggravation: when the network is aggravated by only a single suspect (or a few suspects) over an extended period of time, there is no need for the network to be suspicious of *all* suspects – only those causing the aggravation. Moreover, because the suspect-specific network aggravation towards suspect A in the preceding example will be beyond a maximum threshold, the network has taken steps to block communication with suspect A.

The overall network aggravation may be calculated by taking the average of all the overall target aggravation levels. Thus, in one embodiment, the node maintaining the overall network aggravation level (e.g., server 125) may query each of the targets on the network at predetermined time intervals. Alternatively, each target may automatically communicate its aggravation level to server 125 when its overall aggravation level passes

through a threshold value. In another embodiment, the overall network aggravation is calculated independently of the overall target aggravations.

Throughout the preceding discussion, a specific implementation of network-level reactions at predetermined aggravation thresholds have been described. Depending on the particular configuration, however, different implementations could be employed without departing from the underlying principles of the present invention. For example, while the embodiment described above uses server 125 as the central network repository for calculating and storing network aggravation levels, any node on LAN 105 could provide the same functionality without departing from the scope of the present invention.

Moreover, different mathematical equations may be used to calculate the various threshold triggers. In one embodiment, threshold levels are calculated by combining multiple aggravation variables. For example the equation $A_1 + .75A_{target} + .25A_{network} > T$ can be used to calculate the effective aggravation level using three parts of the overall target aggravation level, one part of the overall network aggravation level and adding this value to the aggravation level associated with the new incident arriving at the target.

Additionally, each incoming incident may increase each of the different aggravation levels described above based on an unlimited number of equations. For example, in one embodiment the equation $A = S*Count_{log}2$ will adjust the aggravation level "A" based on the severity of the incident type "S" and the current incident count "Count."

Throughout the preceding discussion, specific embodiments of the present system have been described as implemented on a network server (e.g., 125) and/or one or more network workstations (e.g., 110). Depending on the particular configuration, however, different implementations could be employed without departing from the underlying principles of the present invention. For example, the server and detection capability could be combined within a hub, switch, firewall (e.g., 115), gateway (e.g., 120), or a promiscuous mode capture device (e.g., node 118 with adapter in promiscuous mode). In this embodiment, the detection system can simultaneously track multiple targets

according to the traffic it observes passing through the device, rather than just the one target it is implemented on. In this implementation, the network aggravation level is calculated from the targets monitored by the device.

Packet-Level Analysis and Filtration

Data transmission across LAN 105 by workstations 110 and server 125 for one embodiment of the present system will now be described with reference to **figure 2A**, which illustrates the different network protocol layers through which data is passed. When a workstation 110 or server 125 on LAN 105 transmits data to another workstation 114 or server 125 on LAN 105 or across network 130 to a node 140 on LAN 145, the data will pass through each of the transmission layers illustrated in **figure 2A**.

The first transmission layer illustrated in **figure 2A** is the physical layer 210. The physical layer 210 represents the actual medium through which the raw digital network traffic flows. For example, the workstations 110 and servers 125 on LAN 105 may be physically connected using numerous different types of media, including coaxial cable, twisted pair cable (e.g., "10 Base-T"), and fiber optic cable. Alternatively, the workstations and servers may be connected via a wireless transmission system (i.e., an RF or infrared transmission system). The layer directly above the physical layer 210 is referred to as the data link layer 220 (a.k.a. the "media access" layer). The data link layer 220 provides the protocol responsible for providing error-free transmission across the physical layer. It accomplishes this by incorporating the data to be transmitted (i.e. data received from the network layer 225) into data frames and then transmitting the frames sequentially across the physical layer 210. "Ethernet" and "Token Ring" are two well known examples data link protocols.

The network layer 225 resides directly above the data link layer 220 and provides network addressing (among other things) for the data to be transmitted. Data is incorporated into network "packets" at the network layer 225, with each packet

containing a source and a destination address in its header. The "IP" portion of TCP/IP, also known as the Internet protocol, is one well known network-layer protocol.

Thus, when a workstation 110 or server 125 receives data which has been transmitted over LAN 105, it receives the data in data link layer frames, each frame containing one or more network layer packets. It then removes the network packets from the data link frames and transmits the packets up through the remaining protocol layers to the application layer 232 (where the application program resides which requested the data or from which data is being requested by another node).

Referring now to **figure 2B**, an analysis module 250 and a filter module 260 which comprise a portion of the intrusion detection and response system are illustrated. The analysis module 250 provides the aggravation level functionality of the present invention. It receives incoming network packets and determines the origin of the transmitted data and the type of data contained in the packet (e.g., ping, login request . . . etc). It then makes a decision as to how to deal with the incoming data based on the current aggravation levels of the target and the network.

For example, referring once again to **figure 4**, if the suspect-specific target aggravation level towards suspect A is above threshold 450 (after the analysis module 250 receives data packets comprising the latest incident from suspect A), then the analysis module 250 may initiate a passive scan of suspect A. If the new incident received from suspect A causes the analysis module to raise the suspect-specific target aggravation level towards suspect A above thresholds 451 or 452, the analysis module 250 may initiate an active scan of suspect A. If the suspect-specific target aggravation level for suspect A reaches threshold 455, the analysis module may require increased authentication from suspect A before providing suspect A access to the target's resources.

Finally, if the suspect-specific target aggravation towards suspect A is above a maximum threshold level, e.g., threshold 455, then analysis module 250 will apply filter module 260 to selectively filter out all data packets received from suspect A. While the

foregoing discussion is focused on the suspect-specific target aggravation level, it should be noted that the analysis module 250 may also react to new incidents based on any of the other aggravation levels previously discussed (e.g., the overall target aggravation, the overall network aggravation, and the suspect-specific network aggravation).

It should be noted that the analysis 250 and filter 260 modules are inserted in the network protocol stack at the packet level (i.e., between the network and data link protocol layers). Thus, every data packet is analyzed before being passed through to the computer on which analysis module 250 and filter module 260 reside. As such, even "stealth" incidents (those designed to evade logging at the transport 230 and/or application 232 layers) are analyzed.

Throughout the preceding discussion, a specific embodiment of packet-level analysis has been described. Depending on the particular configuration, however, different implementations could be employed without departing from the underlying principles of the present invention. For example, while the embodiment described above analyzes network and transport layer connection incidents from packet information, it may instead choose to read the same information from the transport stack. Similarly, instead of recording login failure packets, the system may read those events from the application layer logging and auditing system. Thus, packet-level information can be reconstructed from other logging, auditing, and monitoring subsystems on the target without departing from the scope of the present invention.

Event-Time Rollup

As used herein an "event" is a particular type of data communication (e.g., a ping) sent from a particular suspect to a particular target. An event data structure 800 is illustrated in **figure 8**. The event structure includes an event type field 810 (e.g., ping), a suspect identification field 820, a target identification field 830, a counter field 840, an interval field 850, and a window size field 860. When a target node or server begins passive scanning because one or more of the aggravation levels described above is

beyond a predetermined threshold level, the target node or server may begin to log (e.g., store to hard disk) a record of all incoming events. In order to conserve memory, numerous events may be combined into a single event structure. This procedure, referred to herein as "event-time rollup" will now be described with respect to **figure 7.**

As shown in timeline 710, a window size 860 of time T initially surrounds each event. When the windows of two successive events overlap as shown in timeline 720, the two events are combined into a single event data object 700. Thus, after the second event arrives at t_2 in figure 720, the event data structure 800 will increase it's count data field from 1 to 2, indicating that there were two successive events which overlapped. This method of combining several events into a single data object is done to preserve memory and also to prevent a particular type of attack by a suspect wherein the suspect attempts to disable the target by filling up the target's hard drive will vast amounts of event data.

In addition, the event window of one embodiment may be dynamically expanded and contracted. For example, if events arrive slowly as shown in timeline 730, the event window can be expanded so that future events separated by the same time interval T_1 will be combined in the same data structure 800. For example, the new window may be expanded to twice time interval T_1. Thus, if the first and second events are separated by a 10-minute time interval and the default window T is 15 minutes, then the window will be expanded to 20 minutes. Moreover, if the event rate slowly decreases, then the window will continue to expand to keep up.

Another variable which may effect the event window size is one or more of the aggravation levels discussed above. Generally speaking, events which are classified as more serious (i.e., events that are more likely to be attacks by a suspect based on one or more aggravation levels) will cause the event window to expand. Thus, generally speaking, the default window size for a login failure event will be larger than the default window for a ping event.

In one embodiment, the only window which is considered when determining whether to include successive events in a single data object is the window associated with the *current* incoming event (e.g., the event at t2 in 720 and at t3 in 730). In this embodiment, the only question is whether the previous event (e.g., occurring at t1 at 720 and 730) falls within the window of the new event. Accordingly, in this embodiment, the current window for the new event (i.e., at t2) will generally need to be larger to produce the same effect because the window surrounding t1 will no longer be a factor.

Difference Reports

In addition, periodic reports may be generated which summarize all *new* events which occur during a predetermined time period. For example, **figure 9** illustrates two periods of time, 910 and 920, during which a series of events, 930-960 occur. Period 910 represents a time period for which a report has already been generated. As indicated in **figure 9**, the first event series A 930 represents a series of two or more events separated by an interval I_A. The events included in event series A 930 are all events of the same type, suspect, and target. These events will not be included in the difference report generated at the end of the second time period 920 because event series A contains events which occur in both time periods.

Similarly, event series B 940, 945 will not be recorded because events of the same type, suspect and target occur in both time periods 910, 920. Thus, in one embodiment, the first step in creating a difference report is to determine whether the series of events in question overlap between the two time periods. If they do overlap, then they are not included in the difference report. If they don't overlap, then event data objects in the previous time period are searched to determine if one of the same type, suspect, and target was recorded. If such an event data object is found, then the events are not included in the difference report. Thus, the difference report produces the same result whether or not two events where combined (using the event rollup method) into a single longer-term event, or remained uncombined as two separate events.

By contrast, event C 950 occurs only during the second time period and will therefore show up on the difference report generated at the end of the second period 920.

Event series D will not be included in the difference report generated after the second time period 920 because event series D does not occur during this time period. If a difference report is generated at all for event series D it will be generated at the end of the first time period 910.

The reason for recording only new events (i.e., of the same type, suspect and victim) is that it alleviates the problem of false positives (i.e., identifying an event as suspicious when it should not be identified as such). In other words, an event which occurs during every time period is not generally considered suspicious.

CLAIMS

What is claimed is:

1. An article of manufacture including a sequence of instructions stored on a computer-readable media which when executed by a network node cause the network node to perform the acts of:

modifying an alert variable based on data transmissions originating from one or more suspect nodes;

triggering a first response when said alert variable reaches a first predetermined threshold level; and

triggering a second response when said alert variable reaches a second predetermined threshold level.

2. The article of manufacture as claimed in claim 1 further including the step of triggering additional responses when said alert variable reaches one or more additional threshold levels.

3. The article of manufacture as claimed in claim 1 wherein one of said triggered responses includes a passive scan of one or more of said suspect nodes.

4. The article of manufacture as claimed in claim 3 wherein said passive scan includes the step of recording data transmissions in a log file.

5. The article of manufacture as claimed in claim 1 wherein one of said triggered responses includes an active scan of one or more of said suspect nodes.

6. The article of manufacture as claimed in claim 5 wherein said active scan includes the step of retrieving information about one or more of said suspect nodes including the network address of said suspect nodes.

7. The article of manufacture as claimed in claim 5 wherein said active scan includes the step of determining the network route taken by data originating from one or more of said suspect nodes.

8. The article of manufacture as claimed in claim 1 wherein one of said triggered responses includes said network node requiring increased authentication from any other node before providing access to its resources.

9. The article of manufacture as claimed in claim 8 wherein said increased authentication includes the step of forcing two or more logins before providing access to its resources.

10. The article of manufacture as claimed in claim 1 wherein one of said triggered responses includes the step of blocking incoming data transmissions.

11. The article of manufacture as claimed in claim 1 wherein said alert variable responds differently over time to particular types of data transmissions.

12. The article of manufacture as claimed in claim 11 wherein said alert variable continuously increases in response to the continuous receipt of a particular type of data transmission until the alert variable reaches a predetermined value.

13. The article of manufacture as claimed in claim 12 wherein said particular type of data transmission originating from said suspect node is an invalid login attempt.

14. The article of manufacture as claimed in claim 11 wherein said alert variable initially increases in response to the continuous receipt of a particular type of data transmission and subsequently decreases in response to the continued receipt of said particular type of data transmission.

15. The article of manufacture as claimed in claim 14 wherein said particular type of data transmission originating from said suspect node is a transmission which retrieves information about said network node (e.g., the "ping" command).

16. The article of manufacture as claimed in claim 1 wherein said data transmissions are analyzed by said network node on a network packet level.

17. The article of manufacture as claimed in claim 16 wherein said data transmissions are filtered by said network node on a network packet level.

18. An article of manufacture including a sequence of instructions stored on a computer-readable media which when executed by a network node cause the network node to perform the acts of:

modifying a first suspect-specific alert variable based on data transmissions originating from a first suspect node; and

modifying a second suspect-specific alert variable based on data transmissions originating from a second suspect node; and

triggering a suspect-specific response when either of said suspect-specific alert variables reach a predetermined threshold level.

19. The article of manufacture as claimed in claim 18 including the act of triggering additional suspect-specific responses when either of said suspect-specific alert variables reaches additional predetermined threshold values.

20. The article of manufacture as claimed in claim 18 including the act of modifying an overall alert variable based on said data transmissions originating from each of said suspect nodes.

21. The article of manufacture as claimed in claim 20 including the act of triggering a response towards each one of said plurality of suspect nodes when said overall alert variable reaches a predetermined threshold value.

22. The article of manufacture as claimed in claim 20 wherein said overall alert variable is more responsive to new types of data transmissions than to data transmissions previously received at said network node.

23. The article of manufacture as claimed in claim 22 including the act of initially increasing said overall alert variable in response to data transmissions originating from a particular suspect node and subsequently decreasing said overall alert variable upon continued receipt of said data transmissions from said particular suspect node.

24. The article of manufacture as claimed in claim 18 including the act of communicating each of said suspect-specific alert variables to a network database residing on a server node.

25. The article of manufacture as claimed in claim 20 including the act of communicating said overall alert variable to a network database residing on a server node.

26. An article of manufacture including a sequence of instructions stored on a computer-readable media which when executed by a network server node cause the network server node to perform the acts of:

storing a plurality of suspect-specific alert variables for a plurality of network nodes;

modifying a network alert variable based on the value of each of said plurality of suspect-specific alert variables; and

triggering a network response when said network alert variable reaches a predetermined threshold level.

27. The article of manufacture as claimed in claim 26 wherein said network response includes the act of notifying each of the plurality of network nodes that they should each increase their suspect-specific alert variable towards a particular suspect node.

28. The article of manufacture as claimed in claim 27 wherein said network response includes the act of said network server node initiating a passive scan of a particular suspect node.

29. The article of manufacture as claimed in claim 27 wherein said network response includes the act of said network server node initiating an active scan of a particular suspect node.

30. The article of manufacture as claimed in claim 29 wherein said network response includes the act of blocking all communication between said suspect node and said plurality of network nodes.

31. An article of manufacture including a sequence of instructions stored on a computer-readable media which when executed by a network server node cause the network server node to perform the acts of:

storing a plurality of overall alert variables for a plurality of network nodes;

modifying a network alert variable based on the value of each of said plurality of overall alert variables; and

triggering a network response when said network alert variable reaches a predetermined threshold level.

32. A method comprising:

receiving a first event from a suspect node;

recording said first event in a first data structure having an event count value;

receiving a second event from said suspect node, said second event being of a same type as said first event; and

recording said second event in said first data structure and incrementing said count value if said second event occurs within a predetermined window of time after said first event.

33. The method as claimed in claim 32 further comprising recording said second event in a second data structure having a count value if said second event occurs outside of said predetermined window of time after said first event.

34. The method as claimed in claim 33 wherein said predetermined window of time is increased responsive to said second event occurring outside of said predetermined window of time.

35. The method as claimed in claim 32 wherein said predetermined window of time is modified based on said first or second event type.

36. The method as claimed in claim 35 wherein said window of time is increased for more serious event types and decreased for less serious event types.

37. The method as claimed in claim 36 wherein said event type is an invalid login.

38. The method as claimed in claim 36 wherein said event type is a ping.

39. The method as claimed in claim 32 further comprising generating a report of all new events which occur over a predetermined time period.

40. The method as claimed in claim 39 wherein an event is identified as a new event by:

determining whether said event is included in a single data structure with one or more pervious events received in a time period preceding said predetermined time period;

searching all data structures generated during said time period preceding said predetermined time period if said event is not included in said single data structure with one or more previous events; and

including said event in said report if said event is not identified in any data structures generated during said time period preceding said predetermined time period.

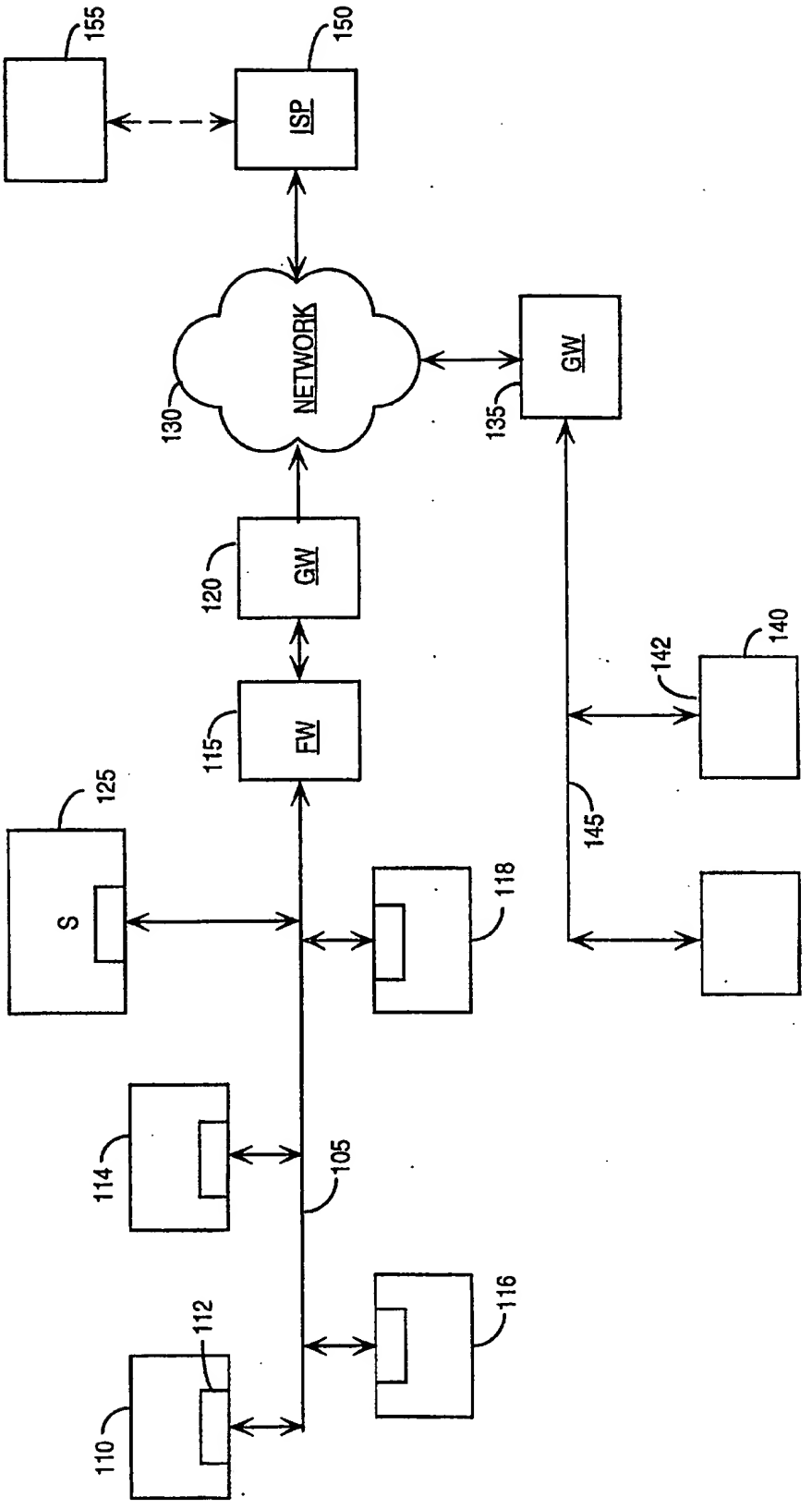

155
150
ISP
NETWORK
130
GW
135
GW
120
FW
115
140
142
125
S
118
145
114
105
116
112
110

FIG. 1

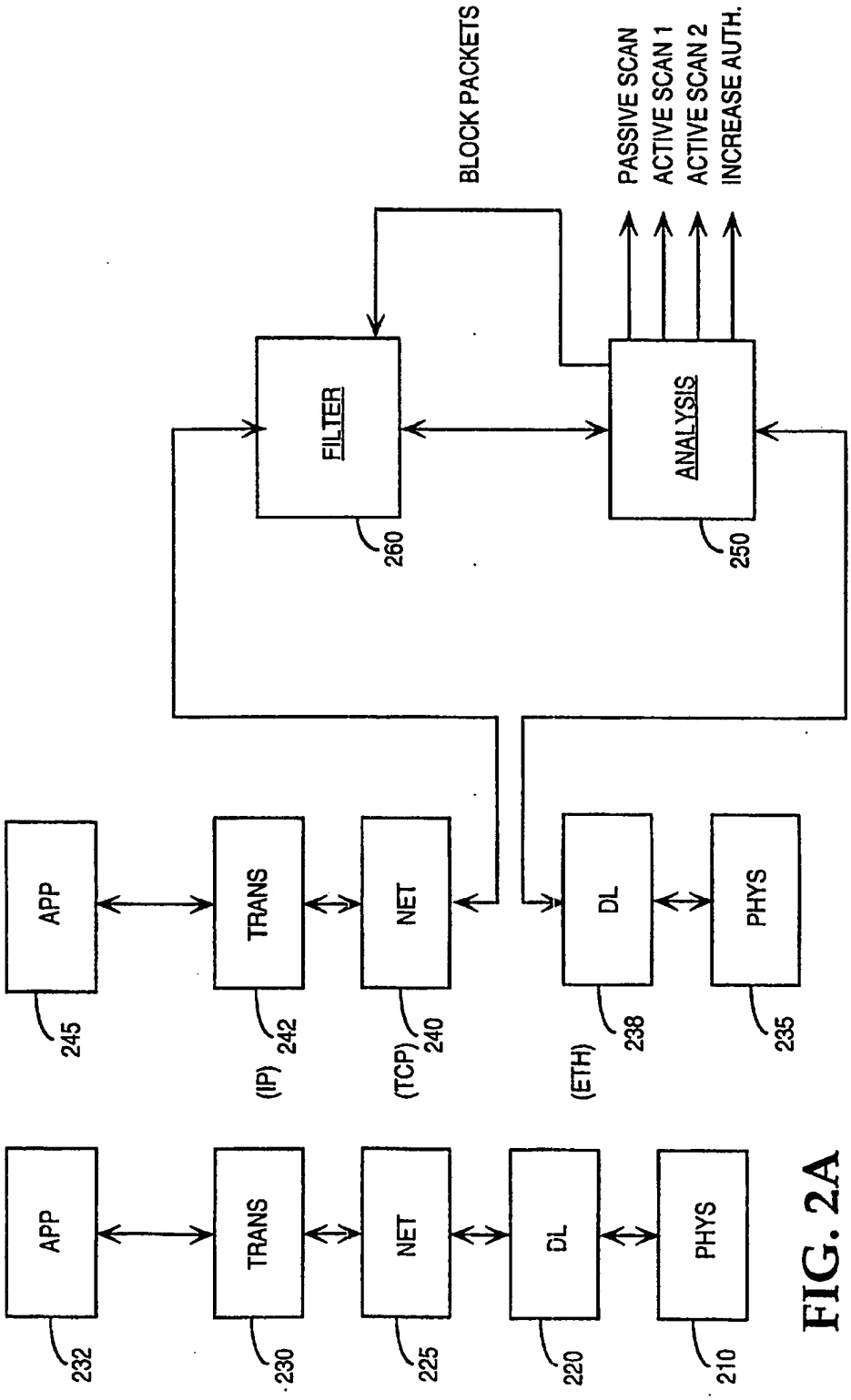

APP
TRANS
NET
DL
PHYS
232
230
225
220
210
APP
TRANS
NET
DL
PHYS
245
(IP) 242
(TCP) 240
(ETH) 238
235
FILTER
ANALYSIS
260
250
BLOCK PACKETS
PASSIVE SCAN
ACTIVE SCAN 1
ACTIVE SCAN 2
INCREASE AUTH.

FIG. 2A

FIG. 2B

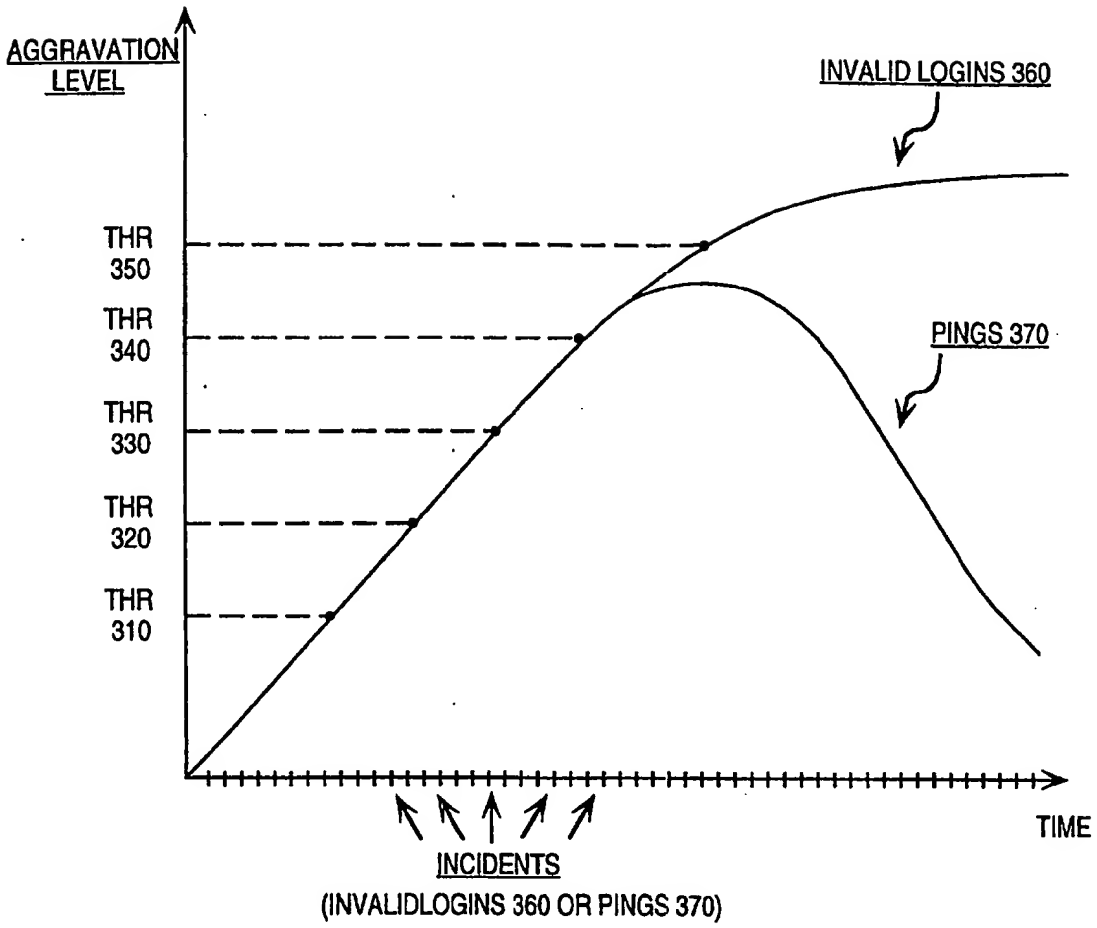

AGGRAVATION LEVEL
INVALID LOGINS 360
THR 350
THR 340
PINGS 370
THR 330
THR 320
THR 310
TIME
INCIDENTS
(INVALIDLOGINS 360 OR PINGS 370)

FIG. 3

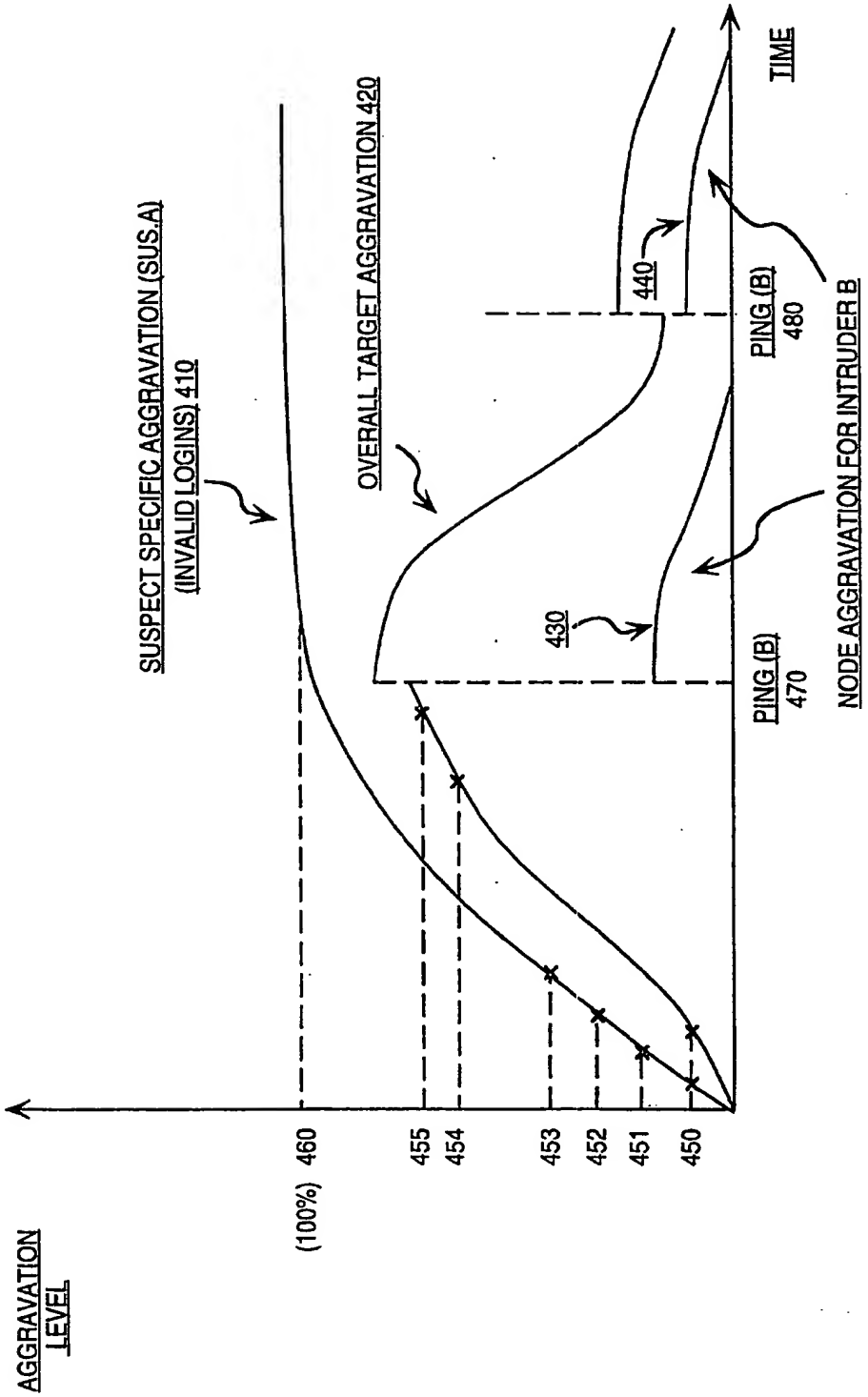

AGGRAVATION LEVEL
(100%)
460
455
454
453
452
451
450
SUSPECT SPECIFIC AGGRAVATION (SUS.A) (INVALID LOGINS) 410
OVERALL TARGET AGGRAVATION 420
430
440
PING (B) 470
PING (B) 480
NODE AGGRAVATION FOR INTRUDER B
TIME

FIG. 4

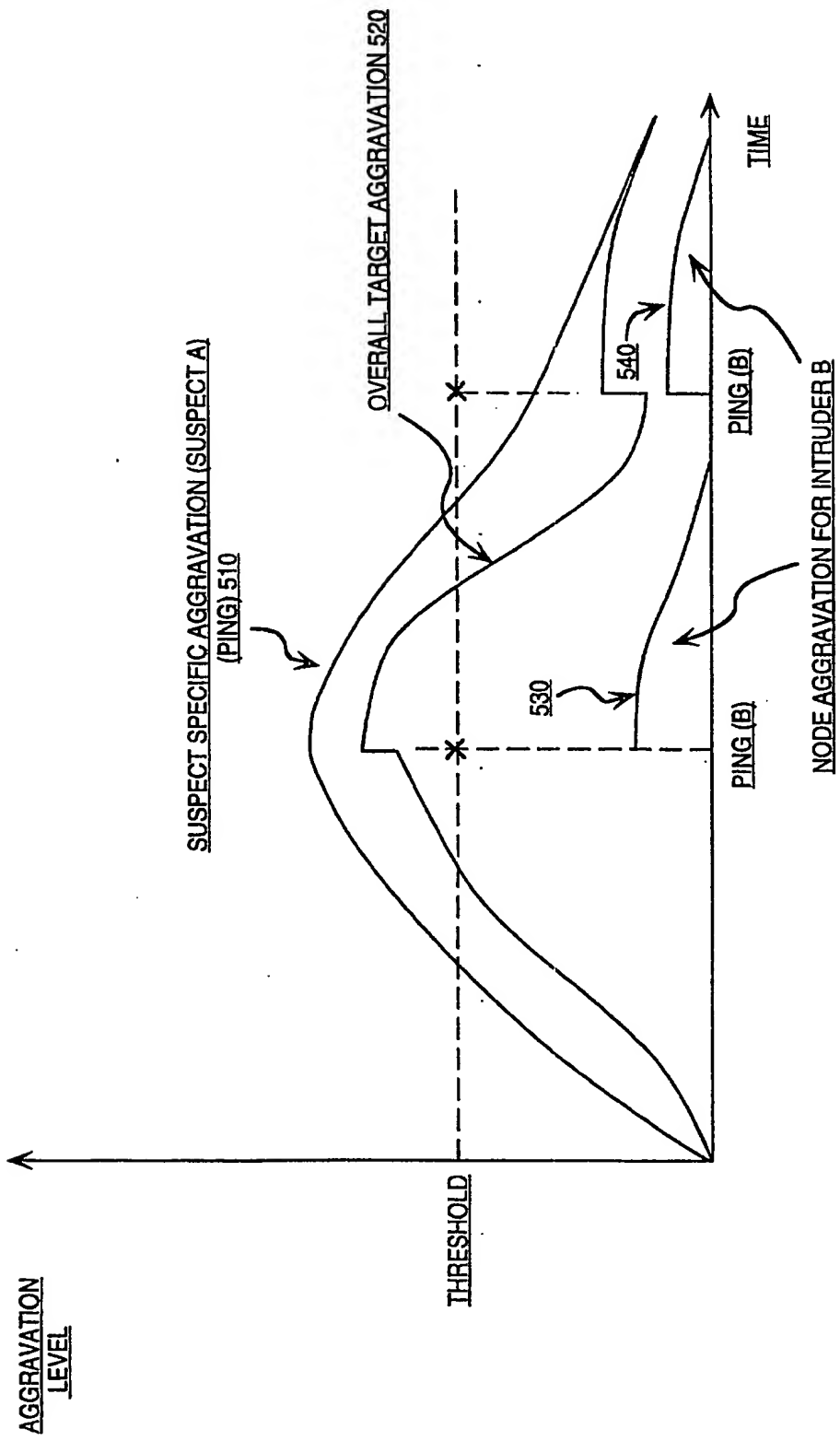

AGGRAVATION LEVEL
THRESHOLD
TIME
SUSPECT SPECIFIC AGGRAVATION (SUSPECT A) (PING) 510
OVERALL TARGET AGGRAVATION 520
NODE AGGRAVATION FOR INTRUDER B
530
540
PING (B)
PING (B)

FIG. 5

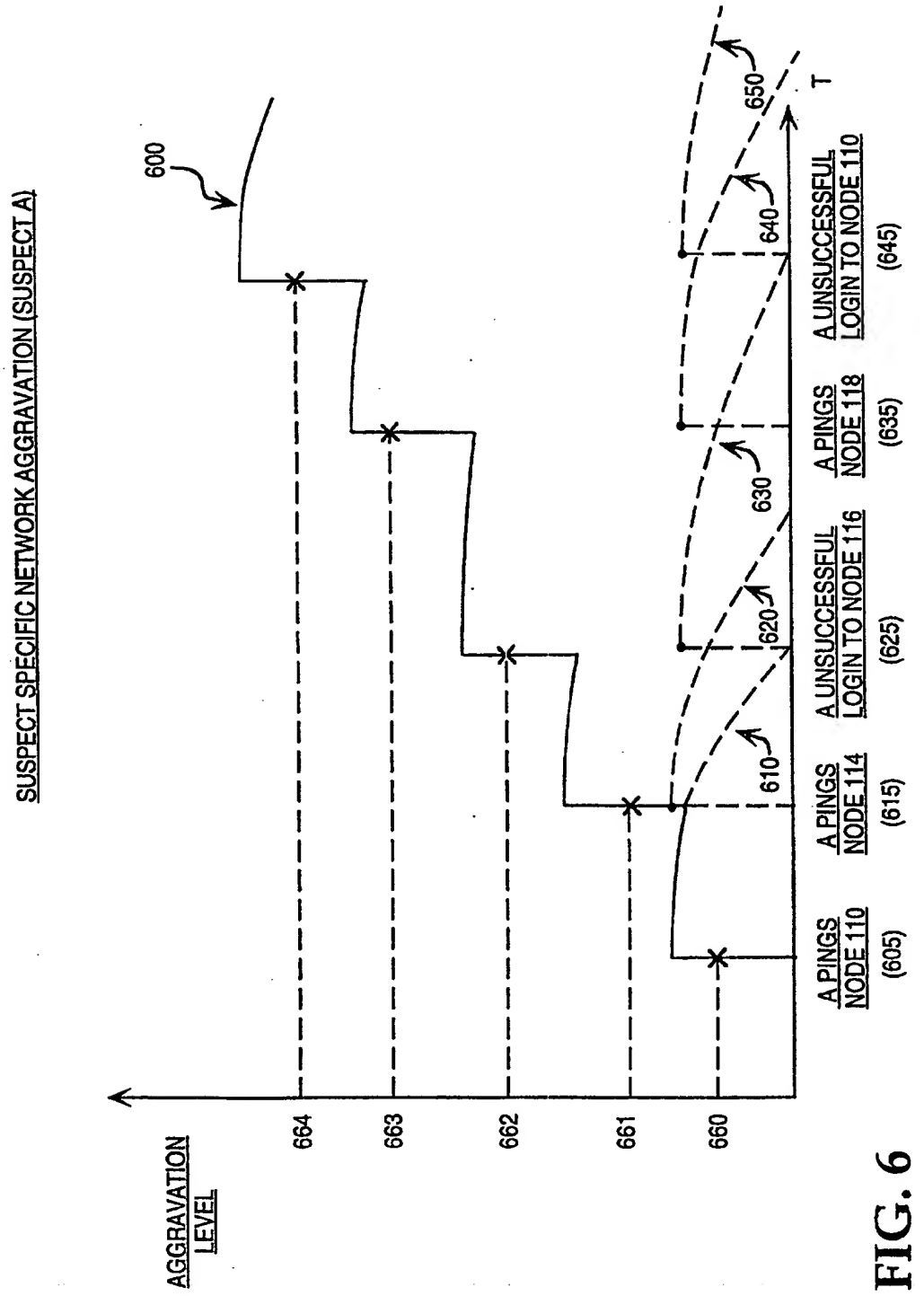

SUSPECT SPECIFIC NETWORK AGGRAVATION (SUSPECT A)
600
650
640
630
620
610
T
A PINGS NODE 110 (605)
A PINGS NODE 114 (615)
A UNSUCCESSFUL LOGIN TO NODE 116 (625)
A PINGS NODE 118 (635)
A UNSUCCESSFUL LOGIN TO NODE 110 (645)
AGGRAVATION LEVEL
660
661
662
663
664

FIG. 6

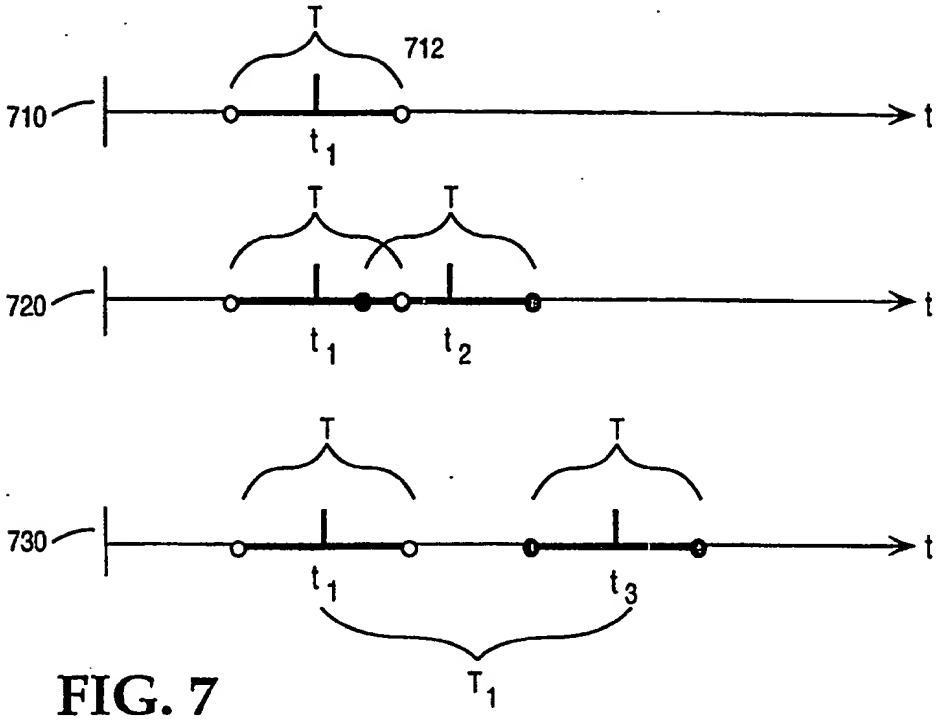

T
712
710
t
t1
T
T
720
t
t1
t2
T
T
730
t
t1
t3
T1

FIG. 7

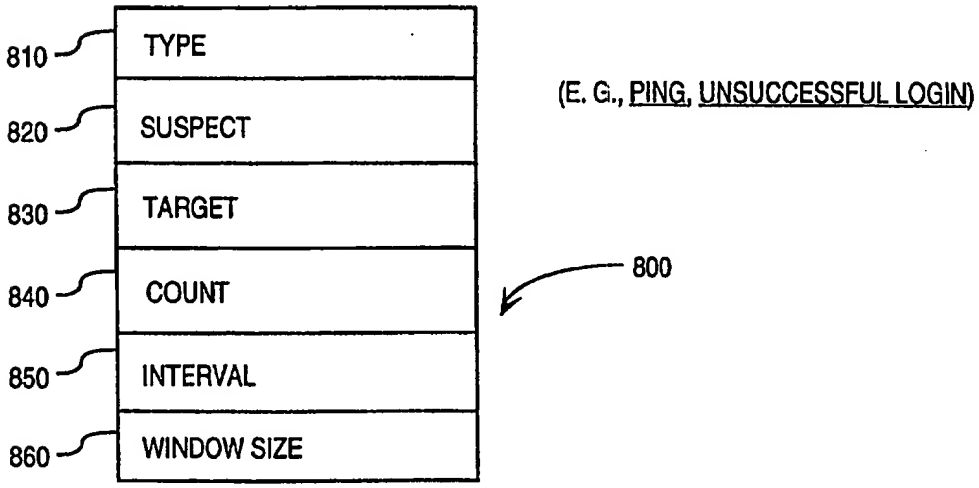

810
TYPE
820
SUSPECT
830
TARGET
840
COUNT
850
INTERVAL
860
WINDOW SIZE
(E. G., PING, UNSUCCESSFUL LOGIN)
800

FIG. 8

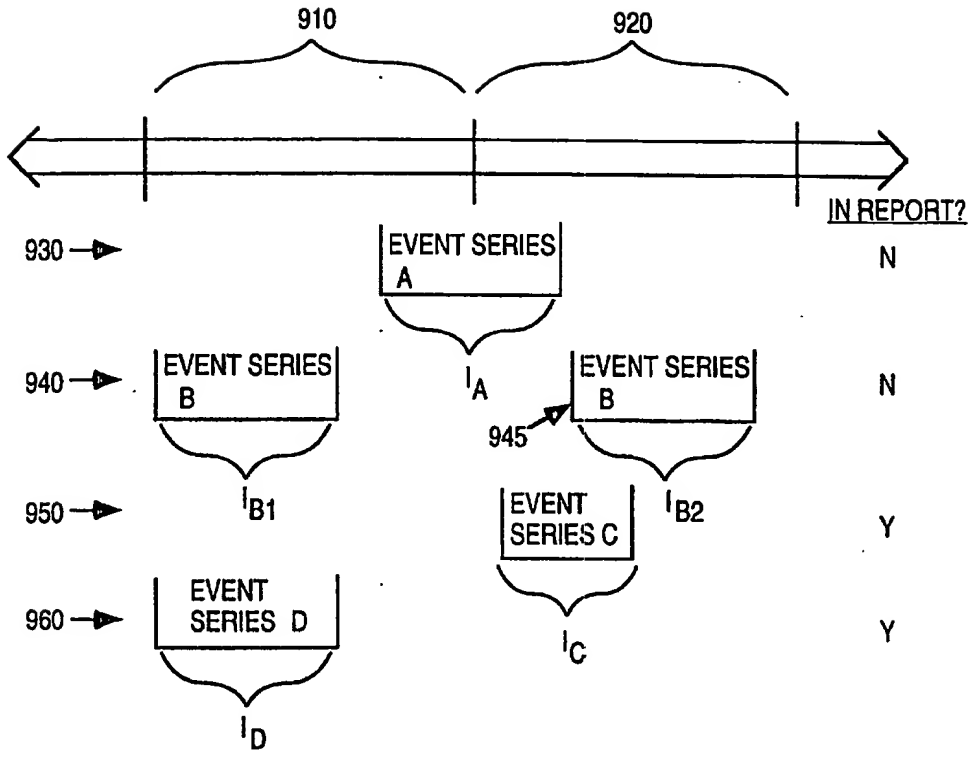

910
920
IN REPORT?
930
EVENT SERIES A
N
I_A
940
EVENT SERIES B
N
945
EVENT SERIES B
I_{B1}
I_{B2}
950
EVENT SERIES C
Y
I_C
960
EVENT SERIES D
Y
I_D

FIG. 9